



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/293,737	04/16/1999	GORDON B. DOW	07844-315001	5291

21876 7590 09/22/2006

FISH & RICHARDSON P.C.
P.O. Box 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER

ANYA, CHARLES E

ART UNIT	PAPER NUMBER
----------	--------------

2194

DATE MAILED: 09/22/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/293,737

Applicant(s)

DOW, GORDON B.

Examiner

Charles E. Anya

Art Unit

2194

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 26 June 2006.
2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-9, 11, 13-23, 28, 29, 31-33, 35, 36, 38-40 and 42-59 is/are pending in the application.
4a) Of the above claim(s) _____ is/are withdrawn from consideration.
5) ☐ Claim(s) _____ is/are allowed.
6) ☒ Claim(s) 1-9, 11, 13-23, 28, 29, 31-33, 35, 36, 38-40 and 42-59 is/are rejected.
7) ☐ Claim(s) _____ is/are objected to.
8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.


Application Papers

- 9) ☐ The specification is objected to by the Examiner.
10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. _____.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.


WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER

Attachment(s)

- 1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____

DETAILED ACTION

1. Claims 1-9,11,13-23,28,29,31-33,35,36,38-40 and 42-59 are pending in this application.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

3. **Claims 1,2,5,6,28,35,42,45 and 46 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 5,469,538 to Razdow in view of U.S. Pat. No. 5,929,864 to Picott et al., and further in view of U.S. Pat. No. 5,404,428 to Wu.**

4. As to claim 1, Razdow teaches a method in a computer program for maintaining dependencies among a set of objects each having a value, the set of objects including an object A and an object B, the method for maintaining dependencies comprising (Numerical dependency graph 16 Col. 8 Ln. 9 - 43) and when the value of object A is a function of the value of object B and the value of object B changes (figure 3A/B Col. 8 Ln. 21 - 43, Col. 11 Ln. 43 - 67, Col. 12 Ln. 1 - 41).

Razdow is silent with reference to marking object A as dirty and not recomputing the value of object A until object A is queried for a value, when the value of object B

Art Unit: 2194

changes, invalidating the dependents of object B and all of their further dependents and severing dependencies among, the dependents of object B and all of their further dependents and causing each invalidated observer-only object to recompute its value by querying the values of the objects from which the observer-only object depends.

Picott teaches marking object A as dirty and not recomputing the value of object A until object A is queried for a value, when the value of object B changes (“...requests...” Col. 7 Ln. 28 – 35).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Picott and Razdow because the teaching of Picott would improve the system of Razdow by providing a seamless interaction by defining a strongly typed and rigidly enforced interface to a set of dependency nodes whereby any dependency node that wants to interact with another dependency node must go through a connection (Col. 1 Ln. 49 – 53).

Wu teaches invalidating the dependents of object B and all of their further dependents, and causing each invalidated observer-only object to recompute its value by querying the values of the objects from which the observer-only object depends (Col. 9 Ln. 1 – 47).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Wu and Razdow because the teaching of Wu would improve the system of Razdow by optimizing evaluation traversal (Col. 9 Ln. 42 - 47). Also see the rejection of claim 11.

Art Unit: 2194

5. As to claim 2, Razdow teaches the method of claim 1, further comprising:
providing object B in the construction of object A, wherein the value of object A is a
function of the value of the object B that was provided in the construction of object A
(figure 3A/B Col. 8 Ln. 21 - 43, Col. 11 Ln. 43 - 67, Col. Ln. 1 - 41).
6. As to claim 5, Wu teaches the method of claim 1, further comprising: when an
object is marked as dirty, breaking any dependency relationships the marked object
may have had; and when the value of an object is recomputed, identifying the objects
on which the recomputed value is actually dependent and identifying the recomputed
object as dependent only on the identified objects (Col. 9 Ln. 1 - 47).
7. As to claim 6, Razdow teaches the method of claim 1, wherein the set of objects
includes settable objects and dependent objects, and each dependent object maintains
a flag whose setting marks the dependent object as valid or invalid or dirty (Col. 9 Ln. 8
- 15).
8. As to claims 28 and 35, see the rejection of claim 1 above.
9. As to claims 42 and 45, see the rejection of claims 2 and 5 respectively.
10. As to claim 46, see the rejection of claim 6 above.

Art Unit: 2194

11. Claims 7,9 29,36 and 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 5,469,538 to Razdow in view of in view of U.S. Pat. No. 5,929,864 to Picott et al.

12. As to claim 7, Razdow teaches a method in a computer program for maintaining dependencies among a set of objects each having a value, the method for maintaining dependencies comprising (Numerical dependency graph (Col. 8 Ln. 9 - 43); identifying the objects upon which a given object depends as those objects into which the given object passed itself as a requester during execution of a compute method of the given object (Col. 8 Ln. 21 - 43, Col. 11 Ln. 39 - 61), and marking the given object as dirty whenever the value of any one of the identified objects changes.

Razdow is silent with reference to not recomputing the value of the given object until the given object is queried for a value.

Picott teaches not recomputing the value of the given object until the given object is queried for a value ("...requests..." Col. 7 Ln. 28 - 35)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Picott and Razdow because the teaching of Picott would improve the system of Razdow by providing a seamless interaction by defining a strongly typed and rigidly enforced interface to a set of dependency nodes whereby any dependency node that wants to interact with another dependency node must go through a connection (Col. 1 Ln. 49 - 53).

Art Unit: 2194

13. As to claim 9, Razdow teaches the method of claim 7, wherein the set of objects includes settable objects and dependent objects, and each of dependent object maintains a flag whose setting marks the dependent object as valid or invalid or dirty (Col. 9 Ln. 8 - 15).

14. As to claims 29 and 36, see the rejection of claim 7 above.

15. As to claim 48, see rejection of claim 9 above.

16. Claims 11,13-18,31,32,38,39,49 and 50-53 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,272,672 B1 to Conway in view of U.S. Pat. No. 5,537,593 to Diamond et al. (prior art being made of record), and further in view of U.S. Pat. No. 5,404,428 to Wu.

17. As to claim 11, Conway teaches a method for changing objects having values defining state of a computer program application, comprising: receiving a change to a value of a changed object, the changed object having objects depending directly on the changed object and objects depending indirectly on the changed object through an object different from the changed object (Col. 21 Ln. 1 - 46), the changed object being a settable object in the compute program application (Section 6.1 Col. 57 Ln. 52 - 57), registering the change with a transaction, dirtying all objects dependent (directly or

Art Unit: 2194

indirectly) on the changed object (Col. 17 Ln. 48 - 67, Col. 18 Ln. 1 - 31, Col. 58. Ln. 12 - 67).

Conway is silent with reference to severing dependencies from the changed object and all of its direct and indirect dependent objects and whenever a leaf object is encountered as a dependent object; enqueueing the leaf object for synchronization after the transaction is committed.

Diamond teaches severing dependencies from the changed object and all of its direct and indirect dependent objects ("...sever..." Col. 8 Ln. 45 – 49).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Diamond and Razdow because the teaching of Diamond would improve the system of Razdow by providing a means for "re-evaluating bounds downward" message is used to propagate downward flowing bounds principally in heterogeneous graphs or trees (Col. 8 Ln. 40 – 42).

Wu teaches whenever a leaf object is encountered as a dependent object, enqueueing the leaf object for synchronization after the transaction is committed (Col. 9 Ln. 1 - 41).

18. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Wu, Diamond and Conway because the teaching of Wu would improve the systems of Diamond and Conway by optimizing evaluation traversal (Col. 9 Ln. 42 - 47).

Art Unit: 2194

19. As to claim 13, Conway teaches the method of claim 11, wherein leaf object synchronization comprises: recomputing a value for each objects marked as dirty, identifying the objects on which tie recomputed value is actually dependent, and identifying the recomputed object as dependent only on the identified objects (Col. 18 Ln. 63 - 67).

20. As to claim 14, Conway teaches the method of claim 13, further comprising: using a requester object to make the transaction consistent, the requester object operating to request an object's value so that the requested value cannot change until the requester terminates, at which time all objects whose values were requested by the requester object are released (Col. 21 Ln. 10 - 19).

21. As to claim 15, see the rejection of claims 11 and 13 above.

22. As to claim 16, Conway teaches the method of claim 15, wherein: the dependency graph represents application state', the roots of the dependency graph are the settable objects of the application state and the intermediate nodes of the dependency graph are dependent objects whose values are the results of intermediate computations (Col. 15 Ln. 46 - 67, Col. 18 Ln. 63 - 67).

23. As to claim 17, Conway teaches the method of claim 15, wherein: the leaf objects of the dependency graph are coupled to a user interface (figure 85 Col. 59 Ln. 44 - 56).

24. As to claim 18, see the rejection of claim 17 above.
25. As to claims 31 and 38, see the rejection of claim 11 above.
26. As to claims 32 and 39, see the rejection of claim 15 above.
27. As to claims 49,50 and 51, see the rejection of claims 13,14 and 16 respectively.
28. As to claims 52 and 53, see the rejection of claim 17 above.
29. **Claims 19-23,33,40,54-59 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 6,272,672 B1 to Conway in view of U.S. Pat. No. 5,526,475 to Razdow and further in view of U.S. Pat. No. 5,689,711 to Bardasz et al.**
30. As to claim 19, Conway teaches a method in computer program for managing dependency among a set of objects, each object of the set of objects having a value, the set of objects including dependent objects, and objects depending indirectly on a given object through other objects of the set of objects (figure 6 Col. 21 Ln 1 - 46).

Conway is silent with reference to each dependent object having a value that is a function of the values of one or more of the other objects in the set of objects, the

Art Unit: 2194

method comprising: calculating the dependency among objects in the set of objects dynamically at the time objects calculate their values and a given object having objects depending directly on the given object.

Radow teaches each dependent object having a value that is a function of the values of one or more of the other objects in the set of objects (), the method comprising: calculating the dependency among objects in the set of objects dynamically at the time objects calculate their values (Col. 4 Ln. 63 – 67, Col. 5 Ln. 1 – 10).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Radow and Conway because the teaching of Radow would improve the system of Conway by providing a method for automatically updating an object tree (Radow Col. 5 Ln. 1 – 10).

Bardasz teaches a given object having objects depending directly on the given object (“...directly dependent...” Col. 20 Ln. 18 – 25).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teachings of Bardasz, Radow and Conway because the teaching of Bardasz would improve the system of Radow and Conway by providing a dependency graph that is created and evaluated in such a way that directed functions and non directed constraint relationships can be intermingled in a single dependency graph (Bardasz Col. 2 Ln. 46 – 51).

31. As to claim 20, Conway teaches the method of claim 19, wherein each observed object in the set of objects has one or more accessor methods that each take a

Art Unit: 2194

requester argument and return a current value of the observed object, the requester argument identifying the object requesting the value of the observed object (Col. 18 Ln. 41 - 67).

32. As to claim 21, Conway teaches the method of claim 19, wherein each settable object in the set of objects has a value setting method that takes two arguments, namely a transaction argument identifying a transaction with which the change to the settable object's value is registered and a new value for the settable object (Col. 58 Ln. 12 - 46).

33. As to claim 22, Conway teaches the method of claim 19, wherein: each object in the set of objects descends from a value class; each computation operation is represented by a Requester object that is owned by a dependent value object, and the Requester object enters the dependent set of one or more value objects from which the dependent value object depends; and the dependent object uses the Requester object to obtain the object values the dependent object needs to calculate its own value.

34. As to claim 23, Conway teaches the method of claim 22, wherein a Transaction class descends from the Requester class, the method further comprising: accumulating changes to one or more settable Value objects in a Transaction object; and executing the Transaction object (Col. 7 Ln. 41 - 52, Col. 8 Ln. 4 - 35).

Art Unit: 2194

35. As to claims 33 and 40, see the rejection of claim 19 above.

36. As to claim 54, see the rejection of claim 20 above.

37. As to claims 55 and 56, see the rejection of claim 22.

38. As to claim 57, see the rejection of claim 23 above.

39. As to claim 58, Conway teaches the method of claim 19, wherein all objects of the set are instantiated from object-oriented programming classes that inherit a set of methods from a common base class (Col. 59 Ln. 58 - 67).

40. As to claim 59, Conway teaches the method of claim 58, wherein the common base class is a requestor class with method to lock down and reset queried values in order to guarantee consistency (Col. 20 Ln. 59 - 67, Col. 21 Ln. 13 - 19).

41. Claims 3,4,43 and 44 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 5,469,538 to Razdow in view of U.S. Pat. No. 5,929,864 to Picott and further in view of U.S. Pat. No. 5,404,428 to Wu as applied to claim 1 above, and further in view of U.S. Pat. No. 6,272,672 B1 to Conway.

42. As to claim 3, Razdow as modified is silent with reference to the method of claim

Art Unit: 2194

1, further comprising: providing in object B a handleReqgest method that adds a requester owned by object A to a dependents list for object B, the dependents list identifying all objects whose value is a function of the value of object B.

Conway teaches to the method of claim 1, further comprising: providing in object B a handleRequest method that adds a requester owned by object A to a dependents list for object B, the dependents list identifying all objects whose value is a function of the value of object B (figure 6 Col. 21 Ln. 25 - 55).

It would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine the teachings of Conway and Razdow because the teaching of Conway would improve the system of Razdow by providing a means of identifying components to be notified of change (Col. 21 Ln. 25 - 55).

43. As to claim 4, Razdow teaches the method of claim 3, wherein the dependents lists for all objects in the set collectively define a directed, acyclic dependency graph (Col. 8 Ln. 9 - 20).

44. As to claims 43 and 44, see the rejection of claims 3 and 4 respectively.

45. **Claims 8 and 47 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Pat. No. 5,469,538 to Razdow in view of U.S. Pat. No. 5,929,864 to Picott as applied to claim 7 above, and further in view of U.S. Pat. No. 6,272,672 B1 to Conway.**

46. As to claim 8, Razdow as modified is silent with reference to the method of claim 7, further comprising: identifying as dependents of a root object all objects that passed themselves as requester objects to the root object or to a dependent of the root object during execution of the requester objects respective compute methods, whereby the set of dependents of the root object is a set that changes based on the computation of dependents and not the root object itself.

Conway teaches the method of claim 7, further comprising: identifying as dependents of a root object all objects that passed themselves as requester objects to the root object or to a dependent of the root object during execution of the requester objects respective compute methods, whereby the set of dependents of the root object is a set that changes based on the computation of dependents and not the root object itself (figure 6 Col. 21 Ln. 25 - 55).

It would have been obvious to one of ordinary skill in the art at the time of the invention was made to combine the teachings of Conway and Razdow because the teaching of Conway would improve the system of Razdow by providing a means of identifying components to be notified of change (Conway Col. 21 Ln. 25 - 55).

47. As to claim 47, see the rejection of claim 8 above.

Response to Arguments

Art Unit: 2194

48. Applicant's arguments filed 6/26/06 have been fully considered but they are not persuasive.

Applicant argues in substance that (1) neither Razdow, Picott nor Wu prior art teach or suggest "when the value of object B changes, invalidating the dependents of object B and all of the their further dependents, including severing dependencies among the dependents of object B and all of their further dependents, (2) the Razdow prior art reference does not teach an object that passes itself as a requestor during the execution of a compute method, (3) the Razdow prior art does not teach or suggest "calculating the dependency among objects in the set of objects dynamically at the time objects calculate their values" and (4) the Diamond prior art does not teach or suggest "severing dependencies from the changed object and all of its direct and indirect dependent objects" because the severing is not performed as a result of a change in object value.

The Examiner respectfully traverses Applicant's arguments:

As to point (1), Applicant concedes that the Wu prior art discloses invalidating dependent derived items when a view model attribute is modified. Examiner contends that the invalidation of dependent derived items includes severing the dependent derived items because Merriam Webster's Collegiate Dictionary (10th. edition) describes invalidate as follows: "to weaken or destroy the cogency...NULLIFY" and the Examiner believes that destroying/nullifying dependent derived items implies severing the dependent derived items.

Art Unit: 2194

As to point (2), for the expression $(x+1).y = 24$ of the column 11 lines 39 – 61 of the Razdow prior art reference, x and y are requestors object. This is because x and y respectively request for the values 5 and 4 that are used to evaluate/compute the expression.

As to point (3), the invention as claimed calculates object dependencies dynamically/automatically when the objects calculate the values. The Razdow prior art teaches this limitation because the dependency graph is calculated automatically when a **new value** is generated.

As to point (4), contrary to Applicant's argument the Diamond prior art does teach severing dependencies as a result of a change in an object value. The pruning is triggered in response to a "re-evaluate upward". The "re-evaluate upward" is message that indicate a change in the offspring node/object. In effect the pruning/severing is triggered in response to an indication of a change in the offspring node/object, thus covering the invention as claimed.

Conclusion

49. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the

Art Unit: 2194

shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

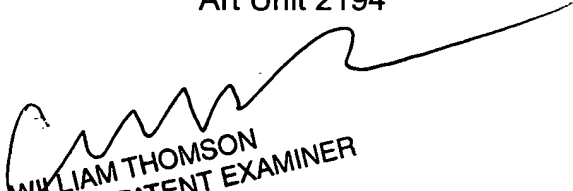
Any inquiry concerning this communication or earlier communications from the examiner should be directed to Charles E. Anya whose telephone number is (571) 272-3757. The examiner can normally be reached on M-F (8:30-6:00) First Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, An Meng-Ai can be reached on (571) 272-3756. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Charles E Anya
Examiner
Art Unit 2194

cea.


WILLIAM THOMSON
SUPERVISORY PATENT EXAMINER